UPDATE on HCAL ELECTRONICS SIMULATION

S.Abdullin, UMD

- QIE instead of amplitude measurement ("a la" ECAL)
- HF treatment splitted from HB/HE
- Noise issue is worrisome
- Some general calo code speed-up possibility(ies).

QIE SIMULATION

- Currently (incl. 4_5_0) an "amplitude reconstruction" using linear combination of ampitude measurements in (-3, +5) time samples.

- After CMS week (5-10 March) a completely different scheme of QIE integration (including ADC quantization) is implemented and tested.

- Evererithing (noise, LSB etc.) expressed in terms of photoelectrons and requires some update (currently used 10 pe / GeV) from J.Elias et al.

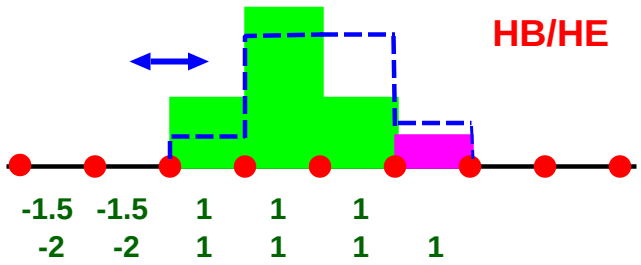
HF SEPARATION from HB/HE

Not straightforward as HF/HB/HE are all in "HCAL"

- if (MyCell().WhichSubDetector() == "HF") is required everywhere

HF and HB/HE divergencies

- photostatistics variation : already taken into account for HF in CMSIM)
- noise : 2 pe (HB/HE) and 0.125 pe (HF)
- shape : short HF signal (peak time ~ 3-4 ns) and longer HB/HE one (32 ns)
- QIE integration : HF signal can be integrated within one time bucket
- QIE sensitivity : LSB = 0.43 pe for HF and 3 pe for HB/HE
- weights : HB/HE = (-2, -2, 1, 1, 1, 1) in "off-line" and (-1.5, -1.5, 1, 1, 1) in TPG,
  HF = (-1, 1) in "off-line" in TPG
- time phase tuning :
  (HF doesn't have proper GEANT time)



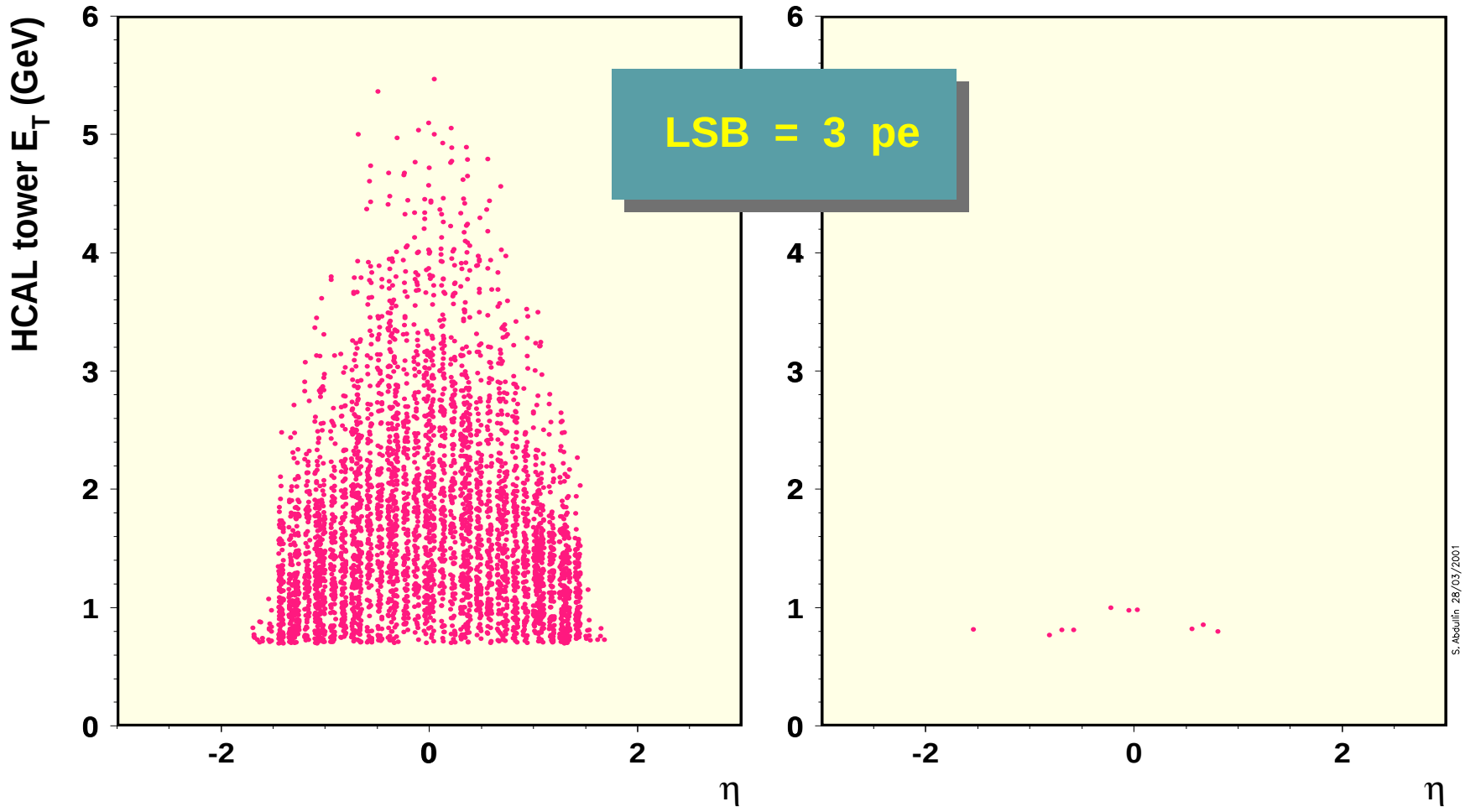
NOISE

- **Noise was small (wrong ?) in ORCA**
  - 0.0006 GeV (GEANT Hit) ~ 45 - 140 MeV of reconstructed energy
  - unique both for HF and HB/HE !
- **Now it increases significantly**
  - HB/HE : 2 pe ~ 240 - 440 MeV per readout (10 pe / GeV)
  - HF : 0.125 pe ~ 200 - 300 MeV per readout (doesn't play a big role)
- **First observations :**
  - HB : excessive number of towers with E_T = 1 - 6 GeV from noise !
  - very few before !

NOISE (cont.)

- noise = 2 pe
- noise = 0.45 pe

(similar to default ORCA value)



SPEEDING-UP the CALO

- **CaloFrontEndResponse :**

  map <CellID, CaloTimeSamples, less<CellID>> samplecache; which is created and deleted 4 times per event (EBRY, EFRY, HCAL, EFSX) and is searched through with smaplecache.find(...);

- **Alternative proposal :**

  - **add unique sequential (transient only) cell number to CellID**
  - **create "static" vector<CaloTimeSample> once for all 4 "Bases"**
  - **samplecache serach and filling with just reference "[ ]"**

- **Memory vs CPU time trade-off**